IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Non Linear Adaptive Filter for Interference Suppression

Bhaskar Gupta[*1], Vinod Kumar Mishra[2]

[*1]Department of ECE BBDIT,Ghaziabad,UP,India

[2]Department of Applied Science, MIT,Partapur Bypass, Meerut, UP, India

bhaskar.pics@gmail.com

Abstract

Radio Frequency (RF) interference is inherent in all wireless systems and is one of the most significant design parameters of cellular and other mobile systems. In this paper, it is shown that how a non-linear adaptive Volterra filter (Polynomial filter where input and output signals are related through Volterra series) helps track the statistics of the input data and dynamics of a direct sequence spread spectrum (DSSS) system. Comparison between LMS (Least Mean Square) and RLS (Recursive Least Square) algorithms, used in filter adaptation process is shown. Results show that adaptive RLS Volterra filter based DSSS receiver is very efficient in suppressing the broadband BPSK interference.

Keywords: Broadband Interference, Non-linear filters, Polynomial filters, Adaptive filtering, Volterra series, least mean square, Recursive least square.

Introduction

Direct sequence spread spectrum (DSSS) systems can operate in the presence of the strong co-channel interference if the processing gain is high enough. However, if this is not the case, or if the interference at the DSSS receiver is very strong, some additional means of the interference suppression has to be implemented [1]. Like in systems, where the interference signal bandwidth is greater or comparable to the system's bandwidth, as in the case of Broadband interference, there the conventional filters fail to provide the clean original signal at the system's output. The inadequacy may be either because of the linearity or even otherwise. In a given situation a linear filter will be optimal only if the signal desired at the output of the filter can be realized by a linear operation on the input signal. It is well known that using the minimum mean square error criterion the optimal filter for estimating a signal d(n) from an input signal x(n) is given by:
$E\,[d(n)|x(n), x(n-1), \ldots, x(n-N+1)]$ where, N is the number of input data points available for the estimation process. If x(n) and d(n) are Jointly Gaussian, this is a linear function in terms of x(n), x(n -1), ... , x(n - N + 1). But, otherwise in general it is a nonlinear function of the input signal x(n). This implies that when x (n) and d (n) are not jointly Gaussian, for optimal processing, one should look for a suitable nonlinear filter. In terms of characterisation in the frequency domain and when the concerned signals can be assumed to be stationary, this condition can be alternately stated as: A nonlinear filter is required when significant spectral components of the input and output signals do not overlap [4]. Secondly, conventional filtering methods does not provide us the desired clean original signal in DSSS system output in presence of Strong BPSK broadband interference if either of the following two conditions exists: The statistical characterisation of the input and desired signals is incomplete or the signal statistics is varying with time. So, here adaptive filtering plays an important role. So, in our problem we plan to use non-linear adaptive filter [9]. In this section we will first discuss the working of Volterra filters further compare the performance of LMS and RLS algorithms so as to select the best adaptive algorithm to be used in Volterra filter for efficient broadband interference suppression and then use RLS Volterra filter for broadband interference excision from DSSS system [5], [6]. Results, expressed as bit error rate (BER), show that the RLS Volterra filter outperforms the LMS Volterra filter in Broadband interference suppression.

Volterra Series Representation

A filter is said to be Volterra if the input and output relation of the filter is given by Volterra series expansion[13]. The Volterra series expansion of a nonlinear system consists of non recursive series in which the output signal is related to the input signal as:

$$y(k) = \sum_{l_1}^{\infty}[w_{01}(l_1)]x(k-l_1) + \sum_{l_1}^{\infty}\sum_{l_2}^{\infty}[w_{02}(\square] l_1, l_2)\,x(k-l_1)x(k-l_2) + ------\infty \ldots\ldots\ldots\ldots (1)$$

Where, $w_{0i}(l_1,l_2,....l_i)$ for $i=0,1.....\infty$ are the coefficients of nonlinear filter model based on Volterra series and Y(k) represents in the context of system identification application, the unknown system output when no measurement noise exists [6]. The underlying assumption is that the kernels $w_0$are symmetric, which means that $w_0(l_1, \ldots, l_n)$ must have the same value regardless of the permutation of $l_1, \ldots, l_n$[11]. If a system has an unsymmetric kernel, Wiener showed that it may be symmetrized by permuting the subscripts on the tiin all possible ways and then taking $w_0$to be 1/n! times the sum of all such w_n. This work assumes the kernels are symmetric [12]. A Volterra filter as can be seen from equation (1) is a particular type of mathematical construct that models many general non-linearities with memory. It provides an input-output relationship, where input and output are either continuous functions or time series in discrete case. A Volterra filter is a polynomial filter which generates output as the weighted sum of product of inputs. From equation (1) one can think of Volterra series expansion as a Taylor series with memory.

As Taylor series has no memory effect it cannot calculate distortion at high frequency as in low frequency analysis but Volterra series reveals an HD2 as high as -32 dB.This Volterra filter characteristic is used in broadband interference detection and suppression [11], [12].

Working of Adaptive Volterra Filter

The principle working of Volterra filter is based on basic adaptive filtering. As explained in detail.

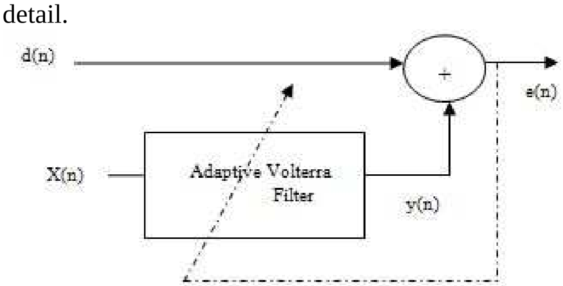

Fig.1 Adaptive filtering block diagram

As the number of iterations increase the error is decreased as the output y(n) approximates the noise estimate in d(n).

Aim of the filter is to produce y(n) very close to d(n) or minimize e(n) such that the interference is suppressed as it reaches to the receiver [9],[10]. According to the above idea the following performance measure used in Volterra filter is defined byMean Square Error (MSE). The MSE is a measure of how the algorithm converges to the true value in a mean square sense &this measurement helps us to see if our system model is indeed minimizing the error and it is sometimes called the learning curve of the algorithm. There can be various orderVolterra filter. But as the order of the filter increases, the complexity also increases. The complexity of a Volterracompensator has been investigated by Tsimbinos.

$$C(N, O) = \sum_{0=1}^{O} (N-1+O)!/(N-1)!(O-1)!$$

Where, N is memory length and O is highest order. Thus in most of the research so far the second and the third order Volterra filters has been used so that the computational complexity can be reduced [7],[8]. As for the sake of simplicity we are using second order Volterra filter for Broadband Interference suppression [2].

DSSS Receiver Model using Volterra Filter

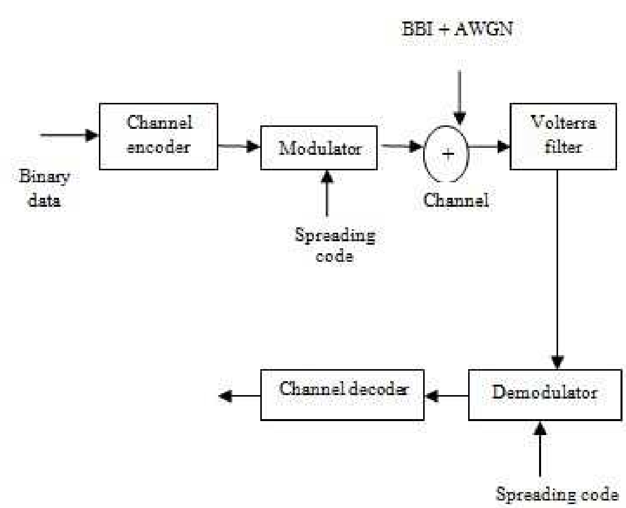

Fig.2 - Block Diagram of DSSS system along with Broadband interference introduced in the channel

figure 2 represents a DSSS communication system where a second order Volterra filter is inserted in between the channel and demodulator. So that the interference and noise that is added in the original signal can be suppressed on passing through the Volterra filter and then the clean signal can be dispread using a p-n(pseudo-noise) sequence at thereceiver circuit. At the output of the channel decoder, we get a clean signal that is our desired signal [3].

Volterra series expansion of second order is described by the truncated version of equation (1):

$$y(k) = \sum_{l_1}^{\infty} [w_{01}(l_1])x(k - l_1) + \sum_{l_1}^{\infty}\sum_{l_2}^{\infty} [w_{02}(\square] l_1, l_2)x(k - l_1)x(k - l_2)$$

Now, the input signal X(n) is interpreted in the following way

$X(n) = [\ x(n)\ \ x(n-1)\ ---\ x(n-N)\ \ x^2(n)\ \ x(n)x(n-1)---\ x(n)x(n-N)\ \ x(n-N)x(n-N+1)\ \ x^2(n-N)]$

The filter coefficients are interpreted in the following way:

$W(n)= [w_o(n)\ \ w_1(n)\ ----\ w_N(n)\ \ w_{0,0}(n)\ \ w_{0,1}(n)\ ------\ w_{0,N}(n)\ ----\ w_{N,N-1}(n)\ \ w_{N,N}(n)$

So, the output, Y(n) is as follows:

$Y(n)=W^T(n)X(n)$ ……….. (2)

Signal Model

Below discussed is the model of input signal, interference signal and noise signal. Our aim is to suppress the interference signal and noise signal at the receiver's output [7,8].

Let us consider a basic received waveform at the receiver:

r(t)= u(t) + i(t) + n(t)

Received signal consist of useful data signal {u(t)}, Broadband interference signal {i(t)} and AWGN noise {n(t)}.

(a) Modeling of Useful Data signal {u(t)}:

u(t) = U.PNS(t)d(t) cos(w0t)

where, U and w0 mark the DSSS carrier and angular frequency, respectively, and PNS(t) is the pseudo noise sequence of chip duration T. Desired signal input power is Pu, and its effective bandwidth is Bu. Message signal is given by d(t) ϵ {+l,-1} with equal probability.

(b) **Modeling of BPSK Broadband Interference {i(t)}:**

$i(t) = U_s d_s(t + \tau) \cos[(w_0 + 2\pi f\Omega)t + \theta]$

Where, Us and f_Ω stand for the interference amplitude and carrier frequency offset to the BPSK carrier. Random data bit delay τ and initial carrier phase θ are uniformly distributed over the [0,T) and [0,2π) interval, respectively. Interference data bit is given by ds(t)ϵ{+1,-1} having equal probability. Its power at the receiver input is Ps and its effective bandwidth is Bs.

(c) Modeling of AWGN noise {n(t)}:

Third component of the received signal is the additive white Gaussian noise (AWGN) with one sided power spectral density. White noise is a sound or signal consisting of all audible frequencies with equal intensity. At each frequency, the phase of the noise spectrum is totally uncertain. It can be any value between 0 and 2π, and its value at any frequency is unrelated to the phase at any other frequency. When noise signals arising from two different sources add, the resultant noise signal has a power equal to the sum of the component powers. Because of the broad-band spectrum, white noise has strong masking capabilities.

The technical specifications of the various system components are discussed in later sections of this paper along with the Simulation results.

Description of Algorithms Used in the Adaptation of Volterra Filters

The minimization of the objective function implies that the adaptive filter output signal is matching the desired signal. Now, this minimization of the performance function that is the mean square value of the error signal produced at the filter output (as seen in fig. 1) is achieved when the adaptive algorithm successfully updates the filter coefficients in such a way so that the filter output adapts the filter input signal.

The Volterra filter adaptive algorithms introduced in this paper for Broadband Interference suppression are discussed in detail.

LMS Algorithm

The well-known LMS (Least Mean Square) algorithm is a sample based algorithm, which does not require collection of data and does not involve matrix inversion.

Though the LMS algorithm has its weakness such as its dependence on signal statistics, which can lead to low speed or residual errors, it is very simple to implement and well behaved compared to the faster recursive algorithms. The most commonly used statistical performance function is the mean square of the error signal. The update part of the adaptive filter adjusts the filter coefficients to minimize the mean square value of the error signal. On achieving the goal, the statistical average (mean value) of the error signal approaches zero, and the filter output approaches the desired signal. When the input x(n) and the desired signal d(n) are stationary signals (from fig.1) , the optimization process leads to the well-known wiener filter. The LMS algorithm is a good example of a practical algorithm based on statistical approach. The exact detail on how much the coefficients are adjusted defines the time it takes to reach the final solution. This time is known as the convergence time.

LMS is one of the most popular algorithms for noise cancellation due to its simplicity of implementation, low computational complexity, and robust behaviour. The computational complexity of NLMS is O(2N)operations per sample (OPS), where one operation is defined as one real multiplication and other is plus one real addition.

The Volterra filter input and output can be compactly rewritten as

$Y(n)=W^T(n)X(n)$

Where, X(n) and W(n) are N most recent inputs and their nonlinear combinations into one expanded input vector and expanded filter coefficients vector respectively.
The error signal e(n) is formed by subtracting Y(n) from the noisy response d(n)

$e(n)=d(n)-Y(n)$

For the LMS algorithm we have to minimize the error

$E[e^2(n)]=E[d(n)-Y(n)]$

The well known update equation for a first order filter is

$H(n+1)=H(n)+ \mu\ e(n)$ |
X(n)…………………………(3)

Where step-size(μ),controls the convergence behavior of the algorithm: the larger the value of μ the faster the algorithm converges, but this would also cause a greater misadjustment (i.e., larger residual error signal) in steady-state. For the algorithm to be stable, ' the step-size must be chosen from 0 to 2. The most common value of the step-size is often taken to be 0.01.There is no tracking factor in this algorithm that is why the tracking capabilities of LMS algorithm is low.

RLS Algorithm

To overcome the drawbacks of LMS algorithm, RLS algorithm is used.

A commonly used deterministic performance function is the weighted sum of the squared value of the previous error signal samples. This actually puts more emphasis on recent observed error samples and gradually forgets about the past samples.

Volterra kernels estimation by the RLS (recursive least square) adaptive algorithm:
The least square error based on the time average

$$J(n) = \sum_{i=1}^{n} \lambda^{n-i} e^*(i,n)e(i,n) \ldots\ldots\ldots\ldots\ldots (4)$$

Where, e^* (i,n) is the complex conjugate of e(i,n)

$$e(i,n) = x(i) - y_N^T(i)w_N(n) \qquad 0 < i < n$$
$$y_N(i) = [y(i), y, \ldots\ldots\ldots, y(i-N+1)]^T \ldots\ldots (5)$$

e(i,n) is the error using the new tap gain at time n to test the old data at time I, and J(n) is the cumulative squared error of the new tap gains on all the old data.
To obtain the minimum of least square error J(n), the gradient of J(n) is set to zero.

$$\frac{\delta}{\delta w_N} J(n) = 0$$

From the previous equations it can be written as:

$$R_{NN}(n)w'_N(n) = p_N(n) \ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

Where $w_N(n)$ is the optimal tap gain vector of the RLS equaliser:

$$R_{NN}(n) = \sum_{i=1}^{n} \lambda^{n-i} y_N^*(i)y_N^T(i) \ldots\ldots\ldots\ldots\ldots (7)$$

$$p_N(n) = \sum_{i=1}^{n} \lambda^{n-i} x^*(i)y_N(i) \ldots\ldots\ldots\ldots\ldots (8)$$

Where $R_{NN}(n)$ is the deterministic correlation matrix of input data of the equalizer $y_N(i)$ and
$P_N(i)$ is the deterministic cross-correlation vector between inputs of the equalizer $y_N(i)$ and the desired output d(i), where d(i)=w(i). To compute the equalizer weight vector w_N, it is required to compute $R^{-1}_{NN}(n)$.
From equation of R_{NN} (n), it is possible to obtain a recursive equation expressing R_{NN} (n) in terms of R_{NN} (n-1).

$$R_{NN}(n) = \lambda R_{NN}(n-1) + y_N(n)y_N^T(n)$$

Since the three terms in the previous equation are all N by N matrices, a matrix inverse lemma can be used to derive a recursive update for R^{-1}_{NN} (n) in terms of the previous inverse, R^{-1}_{NN} (n-1).

$$R_{NN}^{(-1)}(n) = (1)/\lambda\ [R_{NN}^{(-1)}(n-1) - (R_{NN}^{(-1)}(n-1)\, y_N(n)\, y_N^{T}(n)\, R_{NN}^{(-1)}(n-1))/(\lambda + \mu(n))] \ldots (9)$$

Where,

$$\mu(n) = y_N^T(n)R_{NN}^{-1}(n-1)y_N(n)$$

Based on these recursive equations, the RLS minimization leads to the following weight update equations:

$$w_N(n) = w_N(n-1) + k_N(n)e^*(n, n-1) \ldots..(10)$$

Where,

$$k_N(n) = \frac{R^{-1}_{NN}(n-1)y_N(n)}{\lambda + \mu(n)}$$

Where, λ is the weighting coefficient that can change the performance of the equalizer. If the channel is time-variant, λ can be set to one. Usually 0.8< λ<1 is used. The value of λ has no influence on the rate of convergence, but does determine the tracking ability of the RLS equalizers. The smaller the λ, the better the tracking ability of the equalizer. However, if λ is too small, the equalizer will be unstable.

The RLS algorithm uses $2.5N^2 + 4.5N$ arithmetic operations per iteration.

Result and Discussion

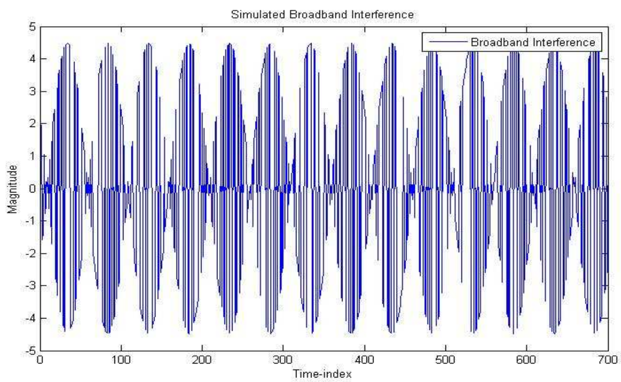

Fig7.(a) Simulated Broadband Interference

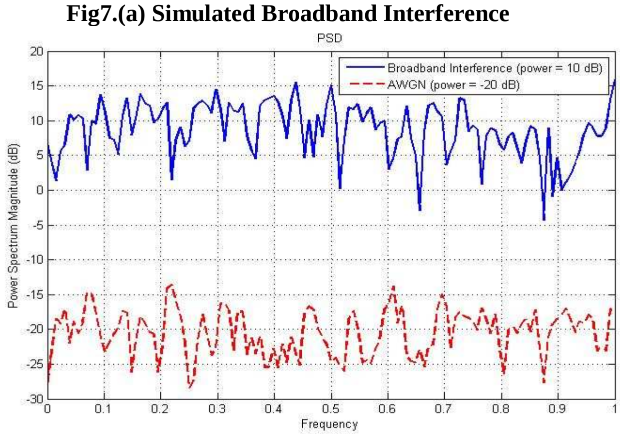

Fig7.(b) Power spectral density (PSD) of simulated Broadband Interference and simulated AWGN

Curves showing the convergence behavior of LMS and RLS algorithm

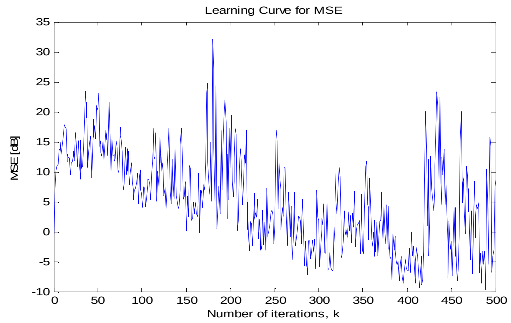

Fig.7(c) the learning curve for the fir systemidentification problem using lms algorithm.

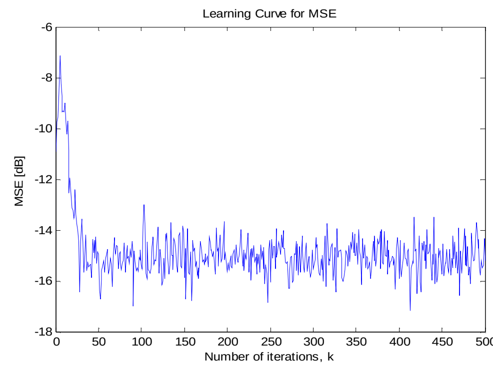

Fig7.(d) The learning curve for the FIR system identification problem using RLS algorithm.

Now, in this section, we examined both adaptive second order Volterra LMS filter (SOVLMS) and adaptive second order Volterra RLS filter (SOVRLS). As is visible from fig.7(a), convergence of LMS Volterra filter is poor due to the fact that convergence factor for the LMS algorithm is bounded by the reciprocal of the product of the number of filter coefficients and input signal power. In the case of Volterra filter the terms of the filter is rarely orthogonal, since they are made up of number of inputs and products of input, same input will be present in many input terms. Because of this the LMS adaptation rate is slow for Volterra filters. Furthermore, in LMS, the kernels were calculated using the normal equation given by:

$$W = R^{-1}P$$

Where, W is the modified Volterra kernel, R is the autocorrelation matrix of input signal and P is the cross correlation vector between the desired user data bit and the received sequence. So, due to the calculation of the inverse autocorrelation matrix the computational complexity is increased. The LMS algorithm adaptation rate is dependent on correlation of the inputs or equivalently the correlation between inputs to the filter taps .The convergence factor is also low therefore we avoid using the LMS algorithm. To overcome the disadvantages of LMS algorithm, RLS algorithm is used in which matrix inverse lemma is used and therefore the need for the afreash calculation of autocorrelation matrix inverse at each point is eliminated.

The graph above shows the square error in dB versus number of iterations during the adaptation process. Simulation results shows, SOVRLS is more feasible as is visible in fig.7(b) for the system identification as compared to SOVLMS. The RLS algorithm typically outperforms the LMS algorithm and is preferred method for updating the coefficients of Volterra filters.

Graph showing the broadband interference suppression after the use of Volterra filter

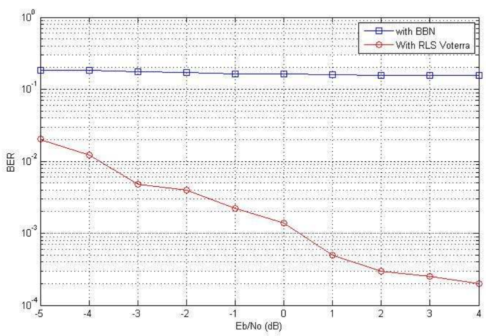

Fig7.(e) Curve showing the variation of BER as a function of E_b/N_o(dB)

This simulation result is obtained for 1000 input bits with spreading code length of 7 and averaged over 100 times and the forgetting factor is taken to be 0.98

As is visible from the above curve that the Bit Error Rate is a decreasing function of E_b/N_O(dB) when RLS Volterra Filter is applied. So, RLS Volterra filter performs satisfactorily in suppressing the Broadband Interference from a DSSS system effectively. The reasons for the efficiency of RLS algorithm is:

- Numerical Stability is very high.
- The rate of convergence is high
- The tracking speed is satisfactory
- It is a modular algorithm

Blue line in the above graph represents the BER vs. E_b/N_o(dB) in presence of broadband interference and red line shows the BER vs. E_b/N_o(dB) curve with Volterra filter. It is seen from the above fig.7(c) that the DSSS receiver containing Volterra filter operate with acceptable BER even in the presence of the interference with bandwidth comparable to the desired signal.Volterra filter provides satisfactory result in broadband interference suppression because it reduces the interference by minimizing the mean square error.

Conclusion

In this paper, most significant work is on a comparative evaluation of the tracking behaviors of LMS & RLS algorithm. Due to high convergence of RLS algorithm, it is much more convenient to use RLS algorithm in Volterra filter. Second most significant work is the DSSS receiver performance with the RLS Volterra filter used for the interference suppression is presented. Interference is modeled as the broadband co-channel BPSK signal with the carrier frequency offset to the desired signal. RLS algorithms are implemented for the nonlinear filter adaptation. Results expressed in BER, show that Volterra filter is successful in the broadband interference suppression process. It enables the DSSS signal reception even in the presence of the strong broadband interference. Apart from broadband interference suppression, Volterra filters have recently gained significant interest in many advanced applications, including acoustic echo cancellation, Channel equalization, biological system modeling and image processing etc. It has been in research since past two decades and is still an active area of research due to its numerous applications.

References

[1] D. Laster and J.H Reed, “Interference Rejection in Digital Wireless Communications”, IEEE Magazine, Vol. 14, No.3, pp.37-62, May 1997.

[2] R Tanner, D.G.M. Cruickshank: “Volterra Filter Structure for DS-CDMA”, Vehicular Technology Conference,VTC’ 96, Atlanta, 1996.

[3] C.E. Davila, A.J. Welch, H.G. Rylander,III: “A Second- Order Adaptive Volterra Filter with Rapid Convergence”, IEEE Trans. on ASSP, Vol.35, No. 6, Sep.1987.

[4] R..Vijayan and H.V. Poor, “Nonlinear Techniques for Interference Suppression in Spread-Spectrum Systems” ,IEEE Trans. on comm, vol. 38, no. 7, July 1990.

[5] D.R. Morgan: Review of the book-Adaptive Signal Processing by Widrow and Stearns. IEEE Trans., Acoustics, Speech, and Signal Processing, Pg. 1017-1018.

[6] R.D. Nowak, “Penalised least squares estimation of Volterra filters & higher order statistics”, IEEE Trans, Image Signal Proc., Vol. 147, No. 2, January 1997.

[7] D. Kocur and R. Zetik, "Volterra Filter Application in DSSS Receiver for Wideband and Narrowband Interference Suppression", IEEE 7th International Symposium on Spread Spectrum Techniques & Applications, pp. 648-651, Czech Republic, 2002.
[8] D. Zoran, D. Miroslav and Z. Vesna, "Broadband Interference Suppression in DSSS Receiver by Volterra Filter," IEEE 5th International Symposium on Spread Spectrum Techniques and Application, pp. 127-129, 1998.
[9] C.F.N Cowan and P.F Adams, "Nonlinear System Modelling: Concept and Applications", Proc. IEEE Int. Conference on Acoustics, Speech,and Signal Processing, pages 45.6.1-4.
[10] P Duvaut, "A Unifying and General Approach to Adaptive Linear-Quadratic Discrete Time Volterra Filtering", Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, pages 1166-1170.
[11] S.P Boyd,"Volterra Series: Engineering Fundamentals", PhD Dissertation, A.B. (Harvard University), 1980.
[12] D.G. M. Cruickshank, "Optimal and Adaptive FIR Filter Receivers for DS-CDMA," International Symposium on Personal Indoor and Mobile Communications, pp. 1339-1343, September1994.
[13] M. Schetzen, The Volterra and Wiener theories of nonlinear systems.New York, USA: JohnWiley and Sons Inc., 1st ed., 1980.